

-10-

REMARKS

Claims 1-40, of which claims 1, 10, 19, 28 and 37-40 are independent, are pending in the application. Claims 5, 6, 14, 15, 23, 24, 32 and 33 have been amended. No new matter is added. All claims are believed to be in condition for allowance.

Claim Rejections Under 35 U.S.C. § 112

Claims 5, 14, 23 and 32 have been rejected under 35 U.S.C. § 112, second paragraph, for reciting the limitation "the base bits", for which there was no sufficient antecedent matter basis. Claims 5, 14, 23 and 32 have been amended to overcome this rejection. Therefore, this rejection should be withdrawn.

Claims 6, 15, 24 and 33 have been rejected under 35 U.S.C. § 112, second paragraph, for reciting the limitation "the smallest range" and "the smallest block size" because of the insufficient antecedent matter basis. Claims 6, 15, 24 and 33 have been amended to overcome this rejection. Therefore, this rejection should be withdrawn.

Claim Rejections Under 35 U.S.C. § 102

Claims 1, 5, 6, 10, 14, 15, 19, 23, 24, 28, 32, 33 and 37-40 have been rejected under 35 U.S.C. § 102(b) as being anticipated by Fox *et al.* (U.S. Patent No. 5,953,522). This rejection is respectfully traversed and reconsideration is requested.


The present invention is directed to a system having a pointer into a segment of memory where the system must be able to determine from the pointer the full bounds of the memory segment. The invention has particular application to pointers which include a permission field which indicates how a process may access data within the segment. As illustrated in Fig. 5, the segment comprises multiple blocks and the address may point to any location in any one of those blocks. As discussed at page 13 with respect to Fig. 4, the base address of a block can readily be determined by setting the B least significant bits equal to zero. However, that approach does not allow the base address of the full segment to be determined. In accordance with the invention of claims 1-36, a finger field in the pointer denotes the block of the segment into which the address

-11-

points. That information, along with the block size and number of blocks in a segment defined by the block and length fields, allows ready computation of the full bounds of the segment. In that way, bounds checking can be performed without using additional memory references to segment bounds.

Fox *et al.* discloses a device driver that provides a functionality of temporary files by representing a virtual device (Abstract, lines 1-4). The device driver includes a device structure that represents the state of the virtual device in which temporary files are stored and maintained. The device structure contains a number of fields, each of which represents a characteristic of the virtual device. One such field is the number of blocks field 408 (see Fig. 4 of Fox *et al.*), which includes data that represents the total number of blocks in the virtual device (Column 11, lines 7-10, 16-18 and 60-62).

When the operating system of Fox *et al.* tries to access data in the virtual device, it looks at a map segment record within a mapping database. Each map segment record includes a base address field, a segment size field, a vnode pointer field, and an offset field. Base address field includes data specifying the first block represented by map segment record. Together, base address field and segment size field specify the range of addresses in terms of an offset from the beginning of the virtual device. (Column 7, lines 35-38, and line 36 through Column 8, line 14).



The Applicants disagree with the Examiner's assertion that Fox *et al.* discloses an address pointing to a segment of memory and a finger field which denotes a block of the segment of memory into which the address points, as recited, for example, in claim 1. The Examiner states that the pointer to a location in memory of Fox *et al.* is comprised of reference 502 of Fig. 5 and reference 408 of Fig. 4. However, nowhere in Fox *et al.* are those two elements combined. Furthermore, the pointer comprising the block length field 408 belongs to an entirely different data structure than the map segment record 502 and describes the total number of blocks in the virtual device, while the map segment record 502 refers to only one of many segments within the virtual device. Therefore, references 408 and 502 are not combinable, and, even taken together, do not represent a pointer to a location in memory.

-12-

The Examiner indicates that an address pointing to a segment of memory is found in Fox *et al.* as reference 508 of Fig. 5. As shown in Fig. 5 and discussed in the specification, reference 508 refers to a vnode pointer field, which includes a pointer to a vnode in vnode layer that represents the state of a segment of anonymous storage, which in turn represents a corresponding segment of the virtual device managed by the device driver. Therefore, the vnode pointer field points, at best, to the beginning of the segment, not into the segment of memory, as claimed in claim 1 of the present application.

In discussing the finger field, the Examiner suggests that the offsets taught in Fox *et al.* are analogous to the finger field. The Applicants respectfully disagree. In Fox *et al.*, offsets are used merely to refer to an address in a segment. As stated in Fox *et al.*, "offset field 510 contains data which specifies an offset into the anonymous storage segment represented by the vnode identified by vnode pointer field 508. Vnode pointer field 508 and offset field 510 collectively specify an offset into a segment of anonymous" (Column 8, lines 22-33, emphasis added). Therefore, both the base address of the segment – the vnode pointer – and the offset are needed to indicate an address pointing to a location in memory; together they are analogous to an address pointing to the segment of memory. As claimed, the finger field is distinct from the address.

Fox *et al.* discloses only a way to address a location in virtual memory and does not teach a finger field that denotes a block into which the address points, as recited in claims 1, 10, 19, and 28, which are therefore not anticipated by Fox *et al.* Dependent claims 5, 14, 15, 23, 24, 32 and 33 depend on the corresponding independent claims and, therefore, are also not anticipated by Fox *et al.* for at least the same reasons as above. Reconsideration is respectfully requested.

Regarding claims 6, 15, 24 and 33, the Examiner states that "Fox discloses the length field is decoded such that the number of blocks is indicated by adding a defined constant (zero) to the value in the length field for all but the smallest range of numbers (zero) and the smallest block size (zero)." The Applicants respectfully disagree. The Examiner does not point to any specific location in Fox *et al.* where such a disclosure is made. Furthermore, the smallest range

-13-

of numbers and the smallest block size refer to the smallest predefined values capable of being represented by the block field and the length field, not to zeros.

Claims 37-40 are directed to front-padded allocation discussed at pages 15-16. As discussed, these claims have application to languages such as Java, where a pointer is always to the first word of an object. Front-padded allocation means that the entire region of memory in which an object is stored is in the end portion of the memory segment. Thus, where an object of size N is allocated into a segment of size M, where M is greater than N, the first word of the object is located at word M minus N of the segment. In that case, the finger field is not required, because it is known that the pointer will always point somewhere into the first block of an object.

Although Fox *et al.* identifies a base address of a segment (Fig. 5, reference 508), it is not the base address of a memory region within the segment, where all portions of the memory segment not within the memory region have addresses less than the address in the pointer. That is, Fox *et al.* does not teach or suggest front-padded allocation. Accordingly, claims 37-40 should be allowed.

Claim Rejection Under 35 U.S.C. § 103

Claims 2-4, 11-13, 20-22 and 29-31 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Fox *et al.* (U.S. Patent No. 5,953,522) in view of Carter *et al.* (U.S. Patent No. 5,845,331). This rejection is respectfully traversed and reconsideration is requested.

It is well accepted that for a claim to be rendered obvious, the prior art reference (or references when combined) must teach or suggest all of the claim limitations. *In re Vaeck*, 947 F.2d 488, 20USPQ2d 1438 (Fed.Cir. 1991).

As discussed above, Fox *et al.* does not disclose the limitations of claim 1. Carter *et al.* also does not teach or suggest the pointer to a location in memory that comprises a block field defining a block size, a length field defining a number of blocks in a segment of memory and a

-14-

finger field which denotes a block of the segment of memory to which the address points. Therefore, the combination of Fox *et al.* and Carter *et al.* does not render claims 2-4, 11-13, 20-22 and 29-31 obvious and the rejection should be withdrawn.

CONCLUSION

In view of the above amendments and remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned.

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By 

James M. Smith

Registration No. 28,043

Telephone: (978) 341-0036

Facsimile: (978) 341-0136

Concord, MA 01742-9133

Dated: 12/2/3